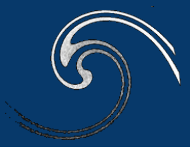
Our Method At a Glance

Remeshing Pipeline:

- Geometry Analysis - input-dependent
  - Parameterization (remove embedding)
  - Geometry Maps (2D images to substitute for 3D)
- **Remeshing Design - realtime**
  - **Flexible Design (use conventional DSP tools)**
  - **Realtime Resampling (use error diffusion)**
- Mesh Generation - output-dependent
  - Triangulation and Reprojection (2D back to 3D)
  - Final Optimization (only if needed!)

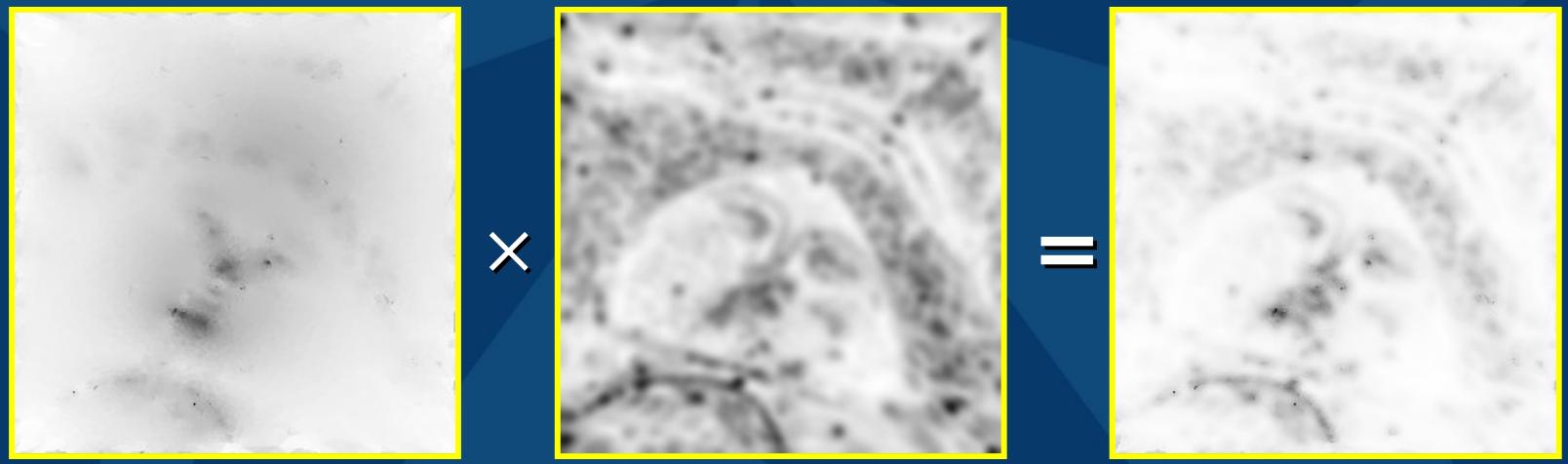


Remeshing Design

Design of the desired vertex density

- Select a sampling criteria
  - Can use any combination of precomputed maps
  - Or any user-defined, spray-painted map
- Multiply (pixel by pixel) by the area map

⇨ **Importance map** (sampling space)

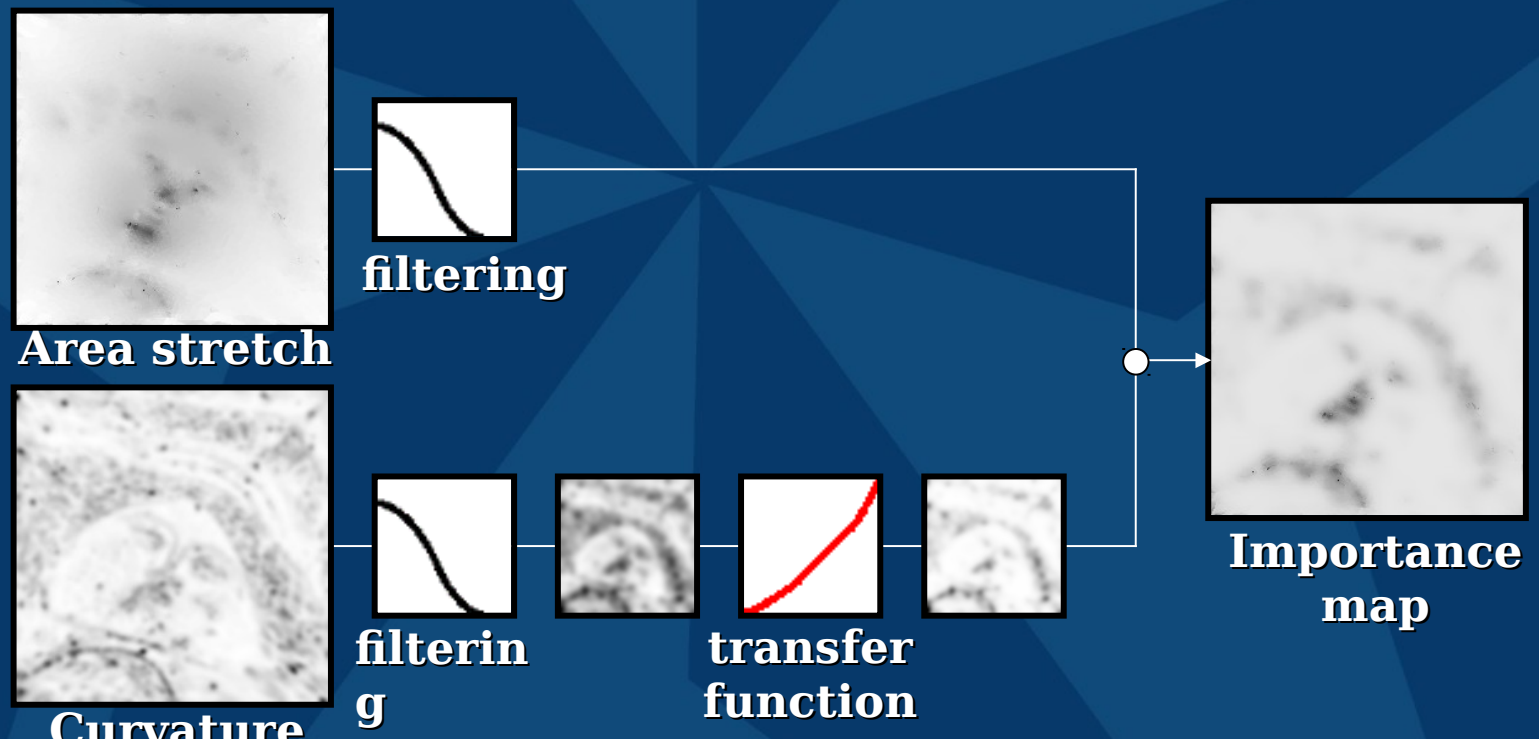


Area stretch × Mean curvature = Importance map

Importance Map Design

DSP for improved control over design

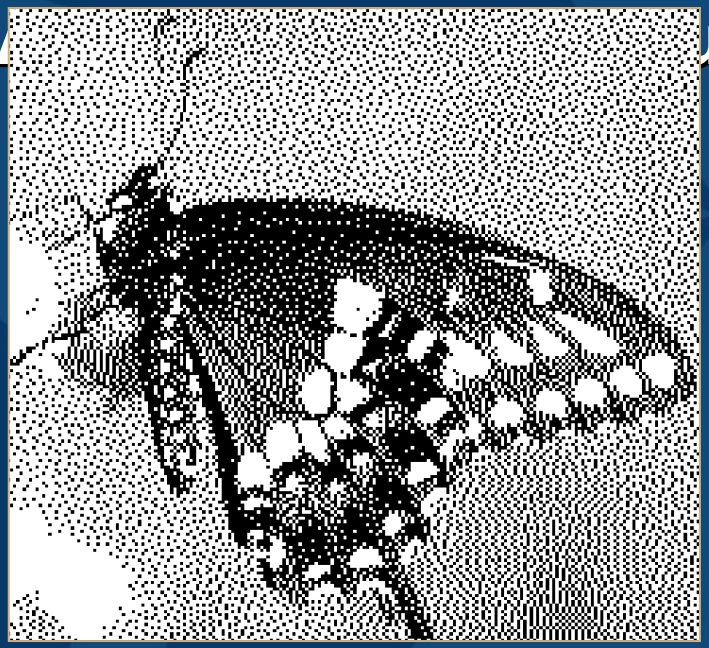
- Filters (e.g., to control mesh gradation)
- Transfer functions (e.g., to tune local density)



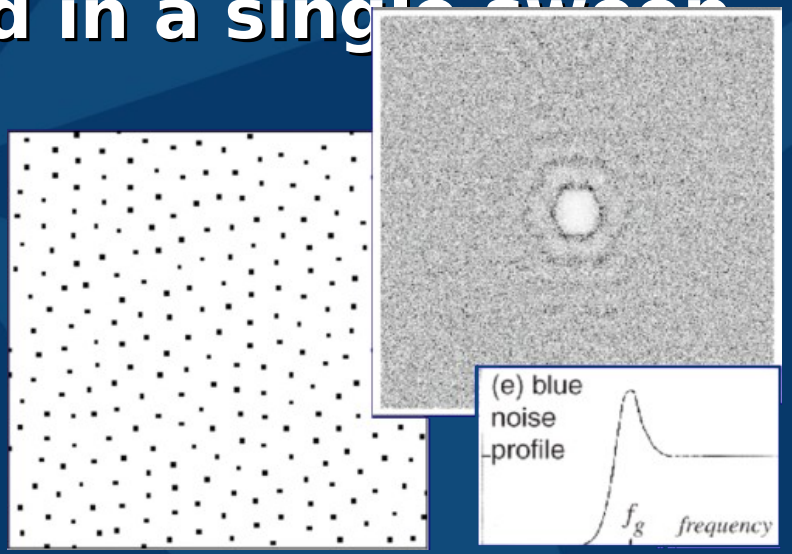
Realtime Resampling

2D Error Diffusion on Importance Map

- Half-toning technique to mimic density
- A[illegible]uted in a sing[illegible]



[Ostromoukhov '01]



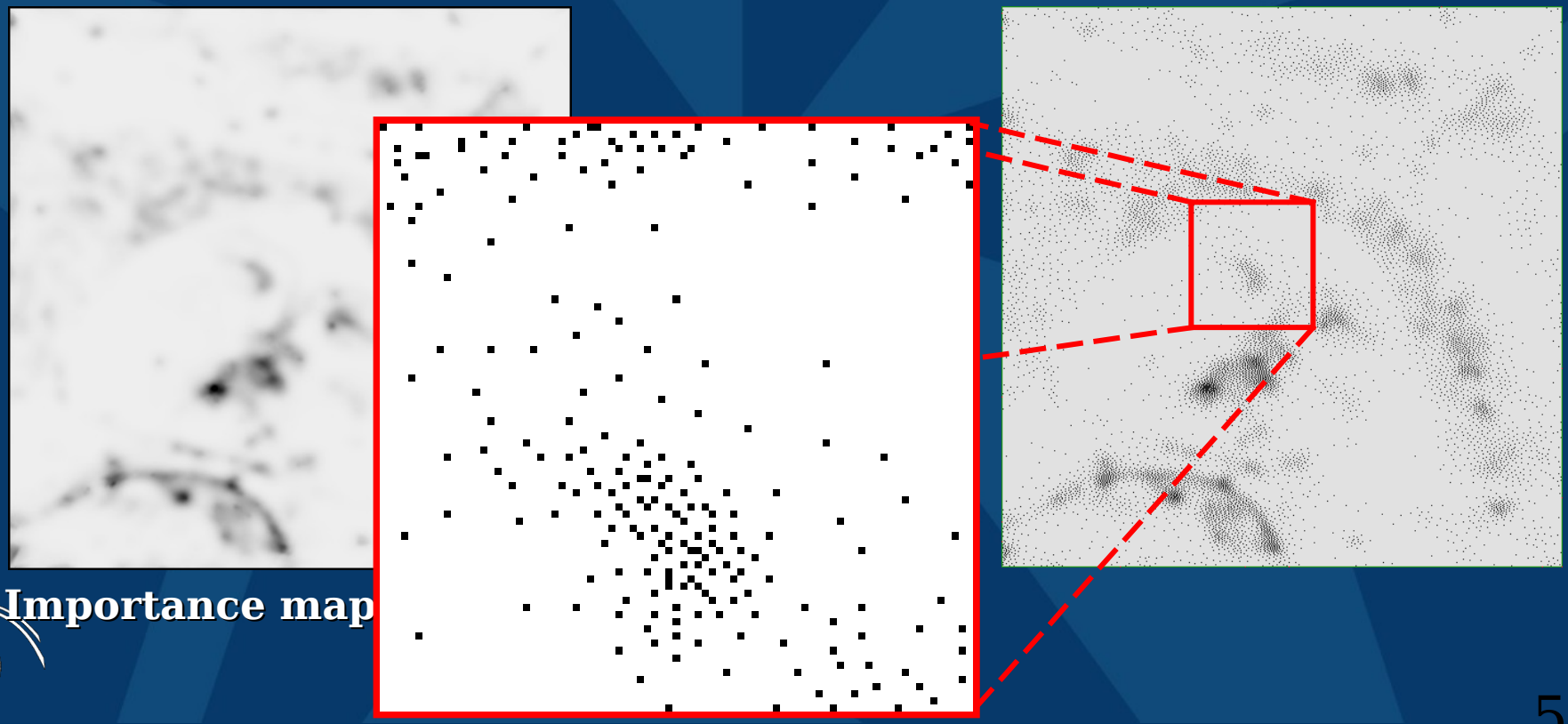
Blue-noise profile, ensuring near optimal placement

Real-time Resampling

Example:

- **512 × 512 picture in 15ms**

  Independent of vertex budget!



Importance map

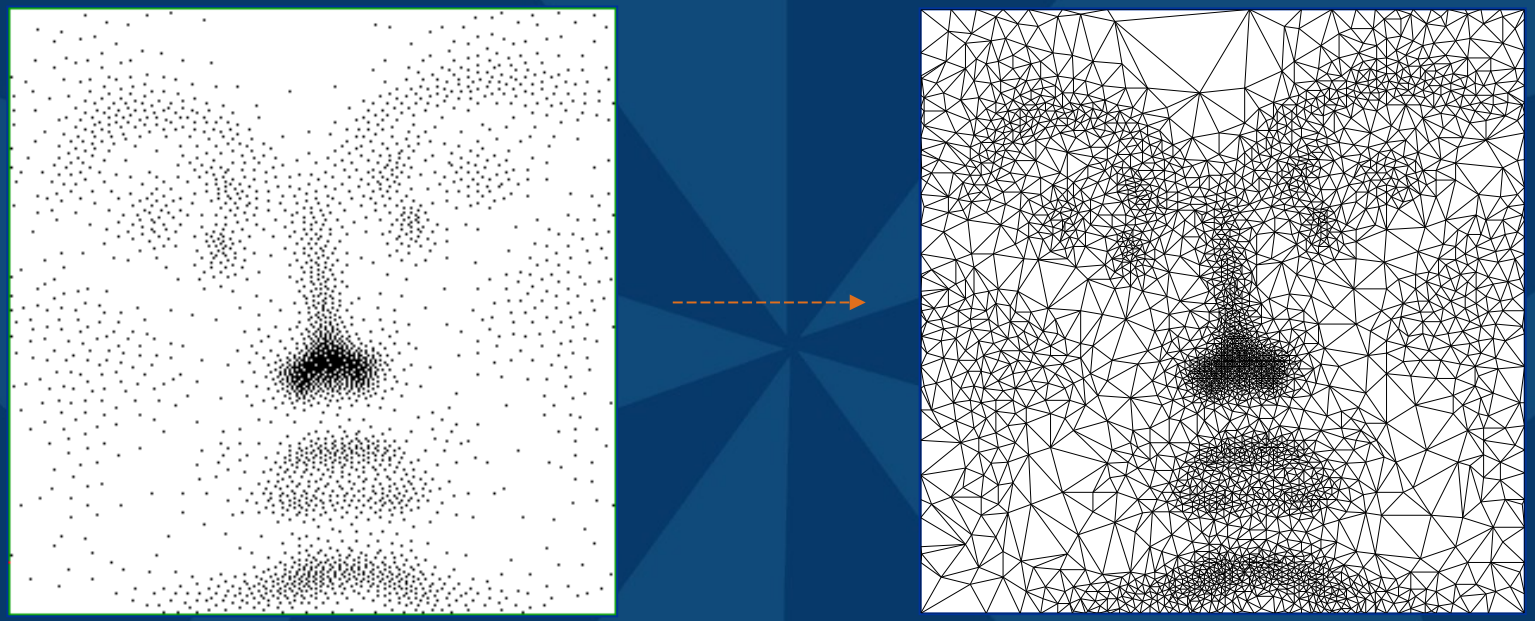
Our Method At a Glance

Remeshing Pipeline:

- Geometry Analysis - input-dependent
  - Parameterization (remove embedding)
  - Geometry Maps (2D images to substitute for 3D)
- Remeshing Design – realtime
  - Flexible Design (use conventional DSP tools)
  - Realtime Resampling (use error diffusion)
- **Mesh Generation** - output-dependent
  - **Triangulation and Reprojection** (2D back to 3D)
  - **Final Optimization** (only if needed!)

Mesh Generation

- **Triangulate in parameter space**
  - Delaunay, for instance [CGAL '02, Shewchuk '97]



- **Connectivity optimization**
  - Swap edges to improve mesh regularity, aspect ratio, etc.

Mesh Generation

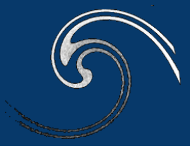
If higher accuracy is needed…

Optimize positions of vertices

- to improve match with importance map
- using weighted Laplacian, for instance

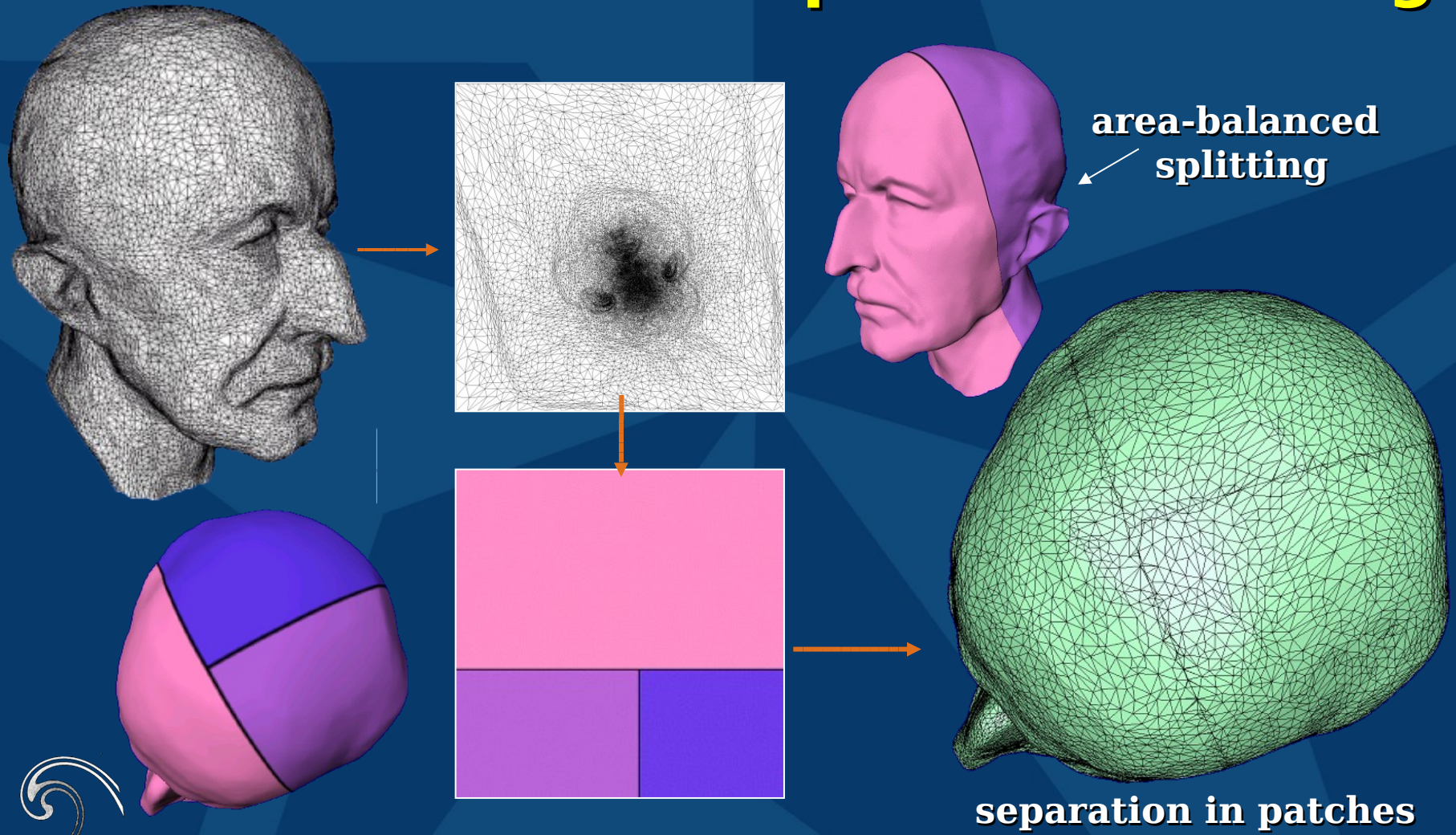
$$\frac{dx_i}{dt} = \sum_j w_j (x_j - x_i)$$

weights are computed using integration over the triangles on 2D maps

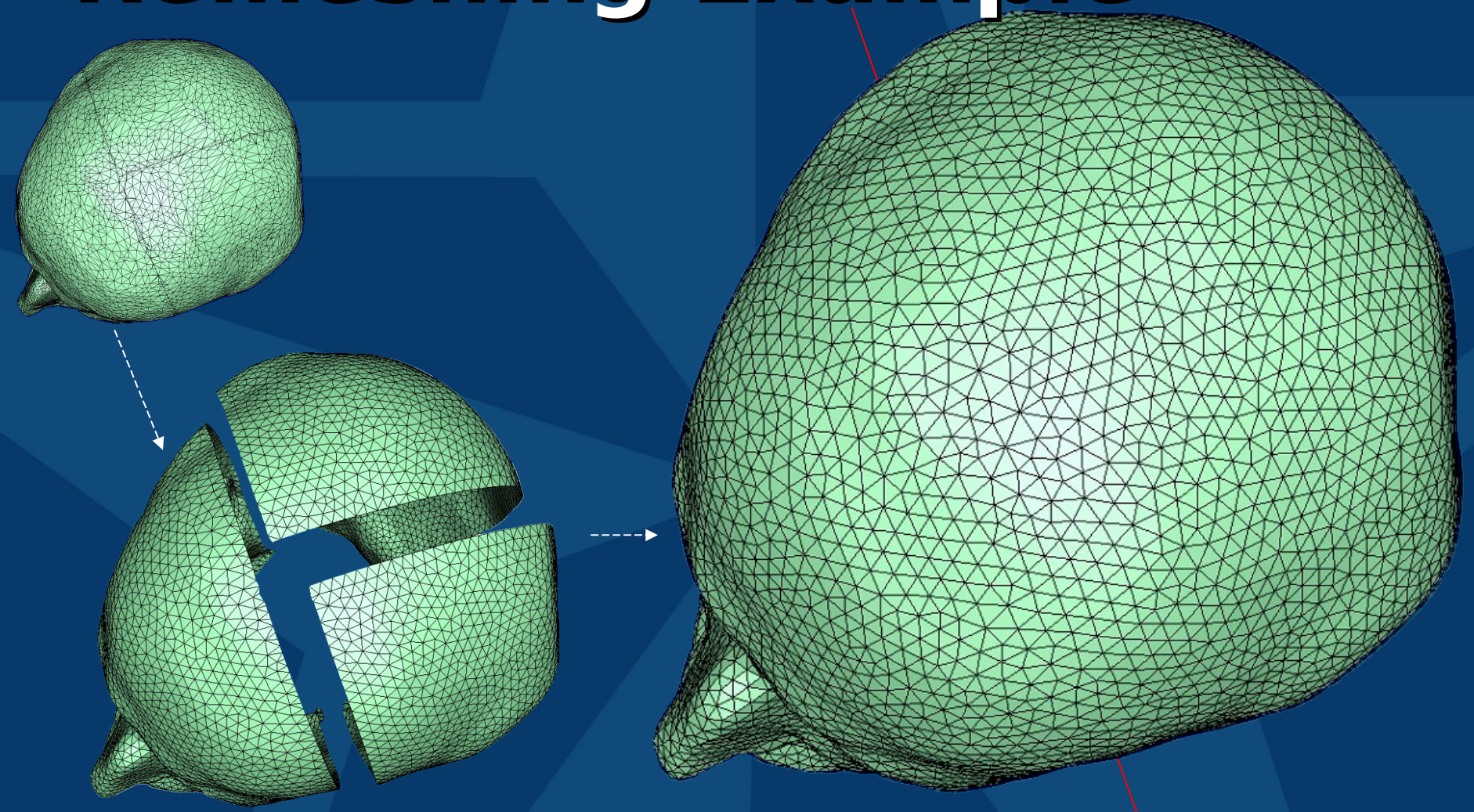


Remeshing Example

The area stretch map can drive tiling



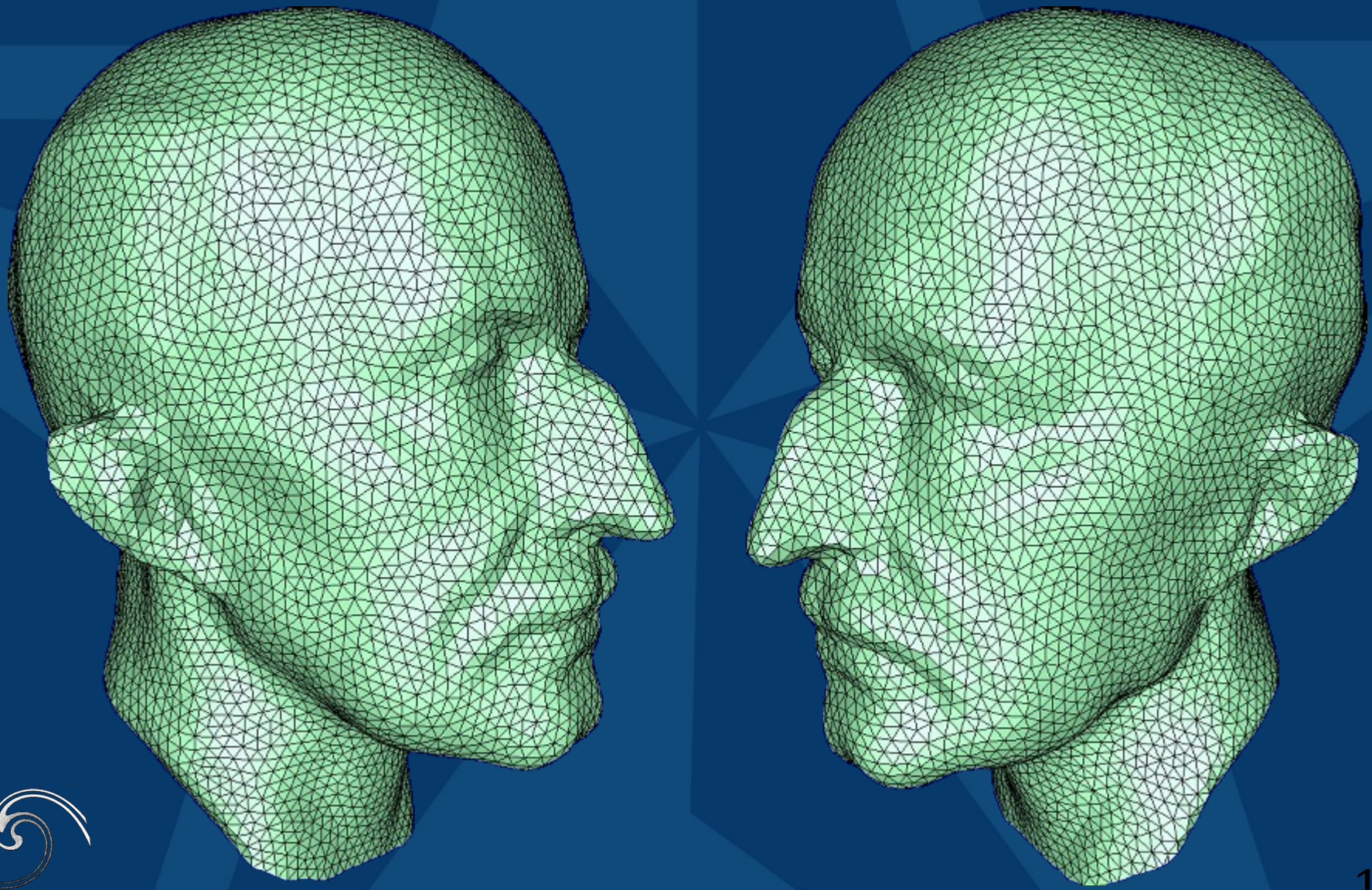
Remeshing Example



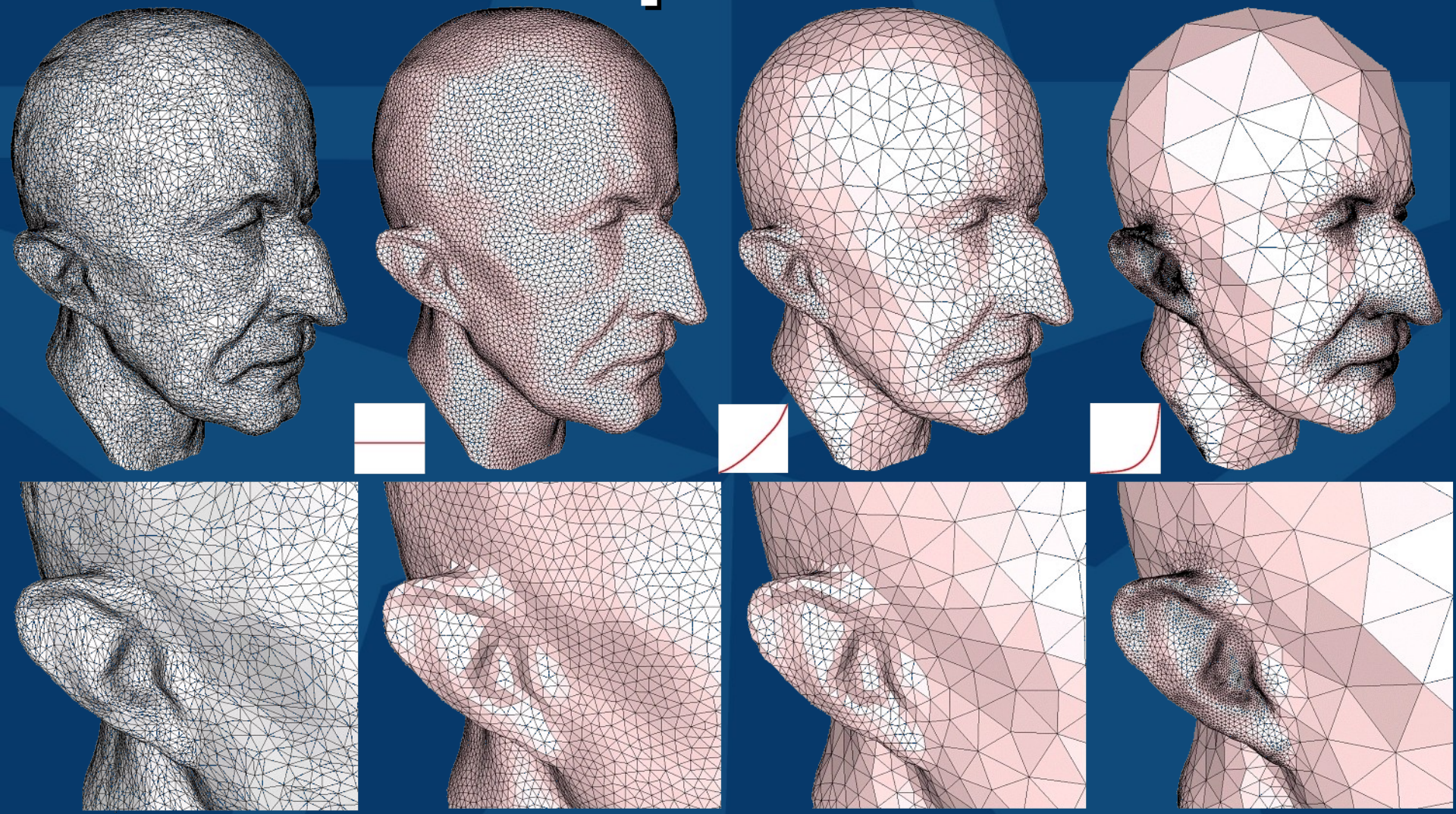
Remeshing patch by patch. Add tangential smoothing.

Stitch together. Soften tangential smoothing further.

Remeshing Example



More Examples

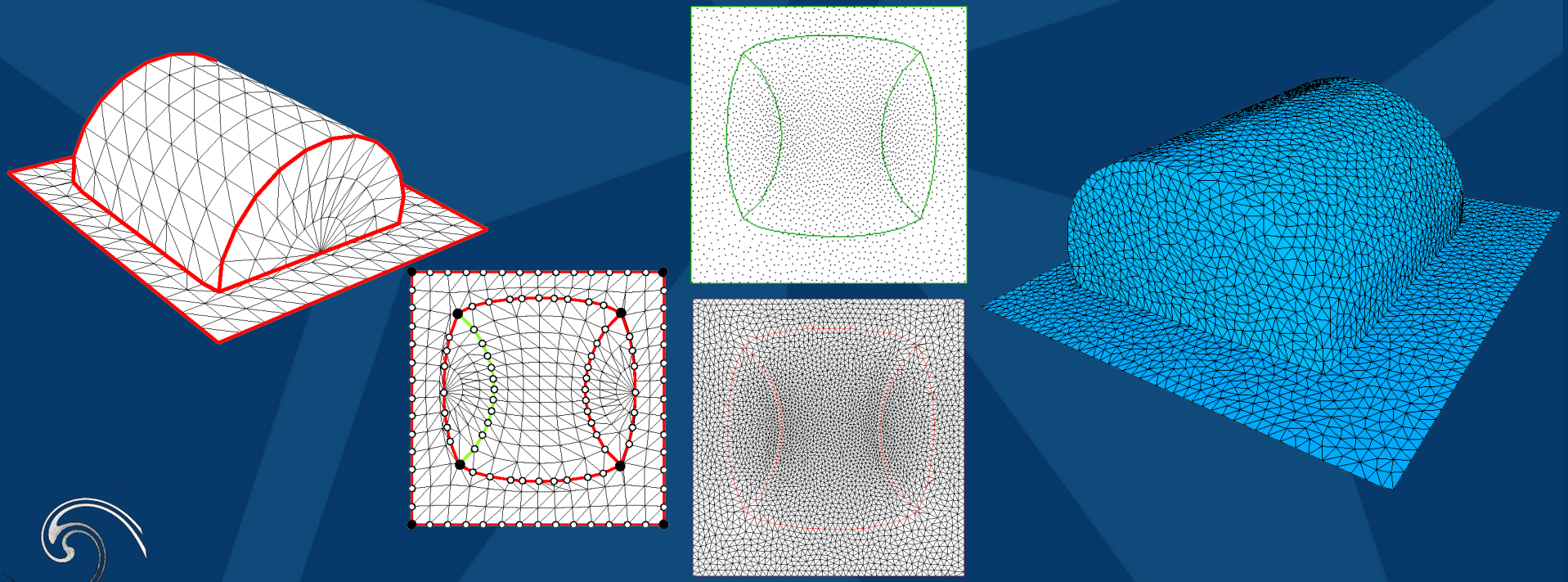


original uniform adapted I adapted I

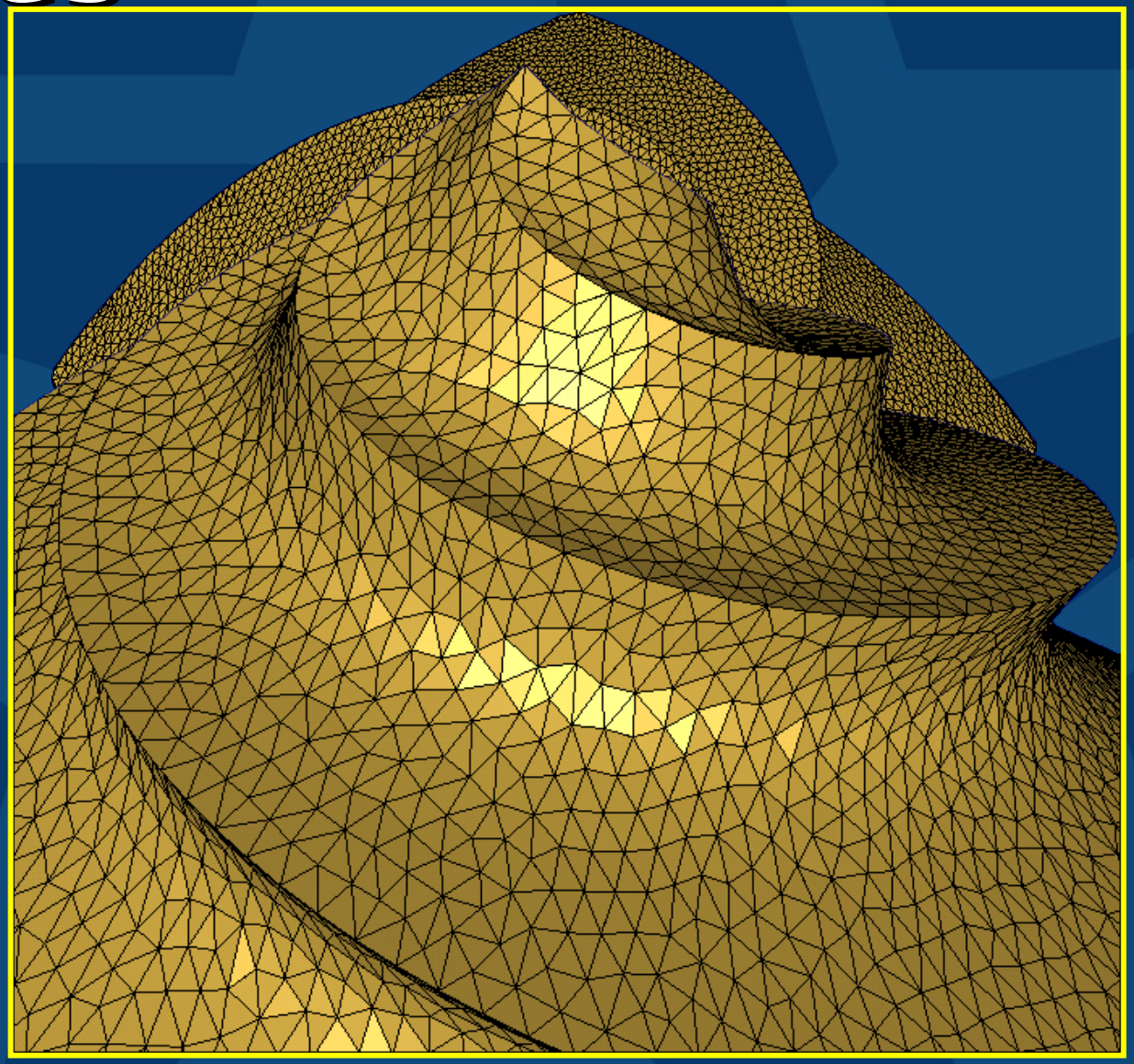
Preserving Features

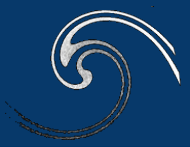
Using a *Feature Skeleton*

- Extract feature graph
- 1D error diffusion along features
- Constrained Delaunay triangulation

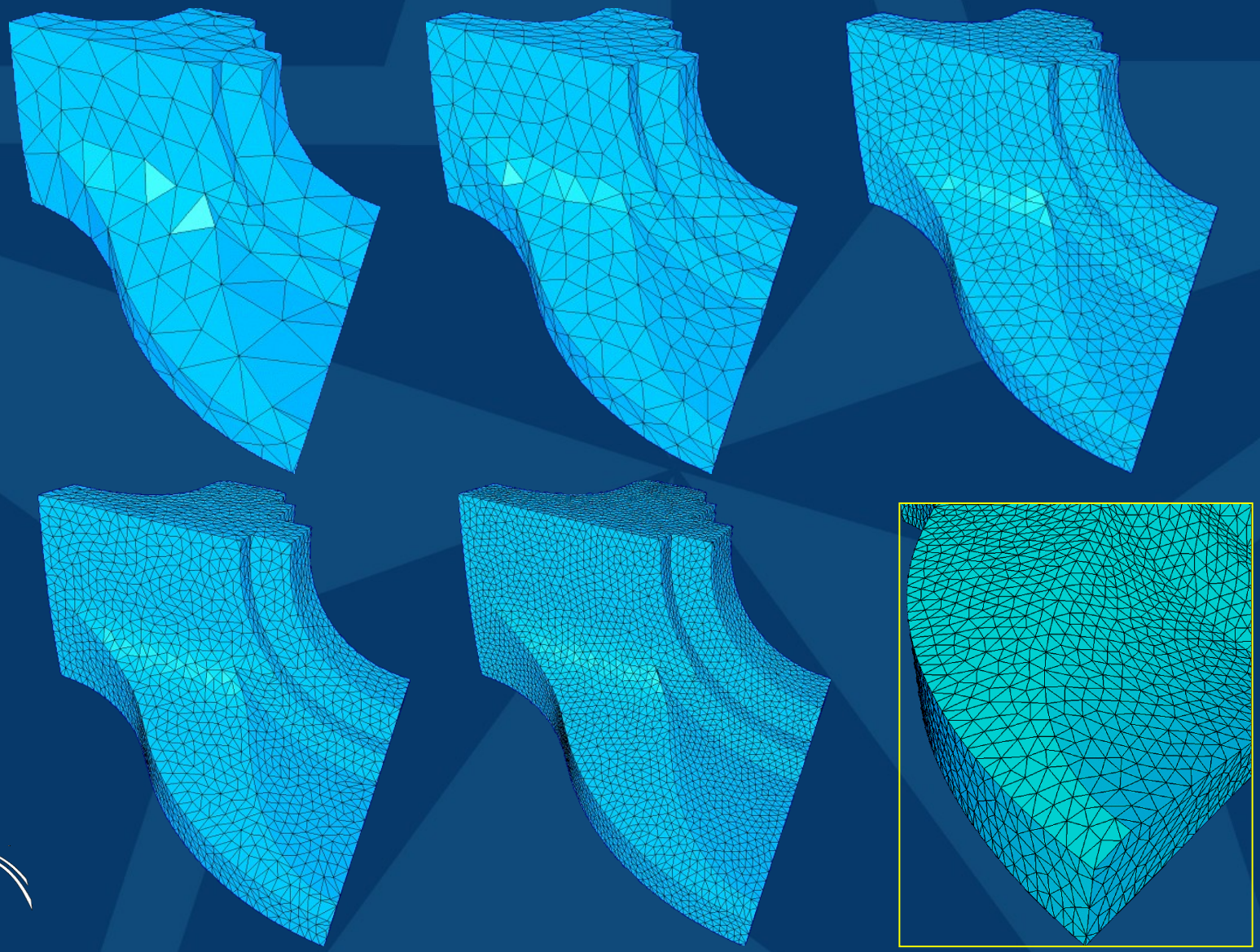


Example With Sharp Edges

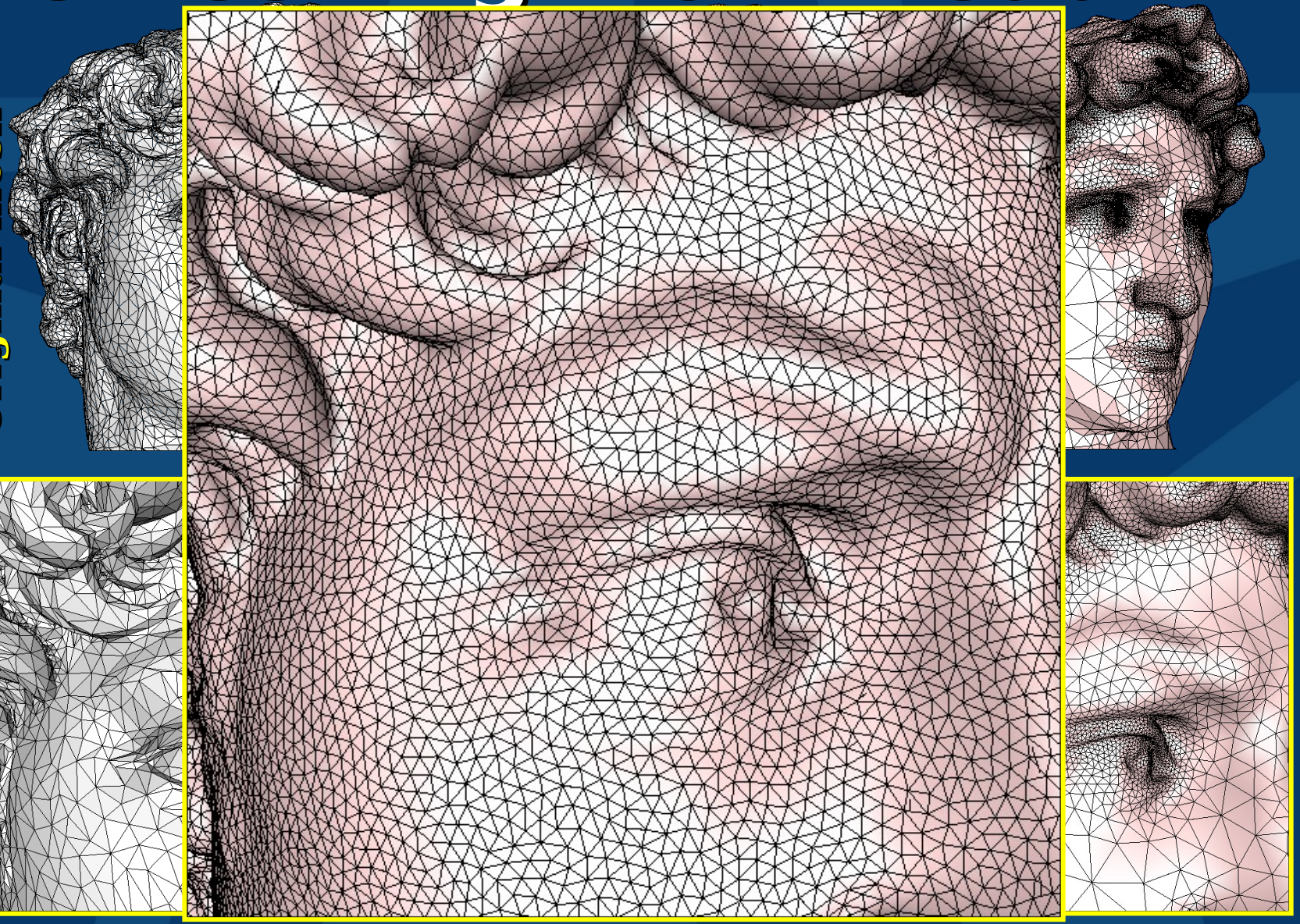




Fandisk Continuum

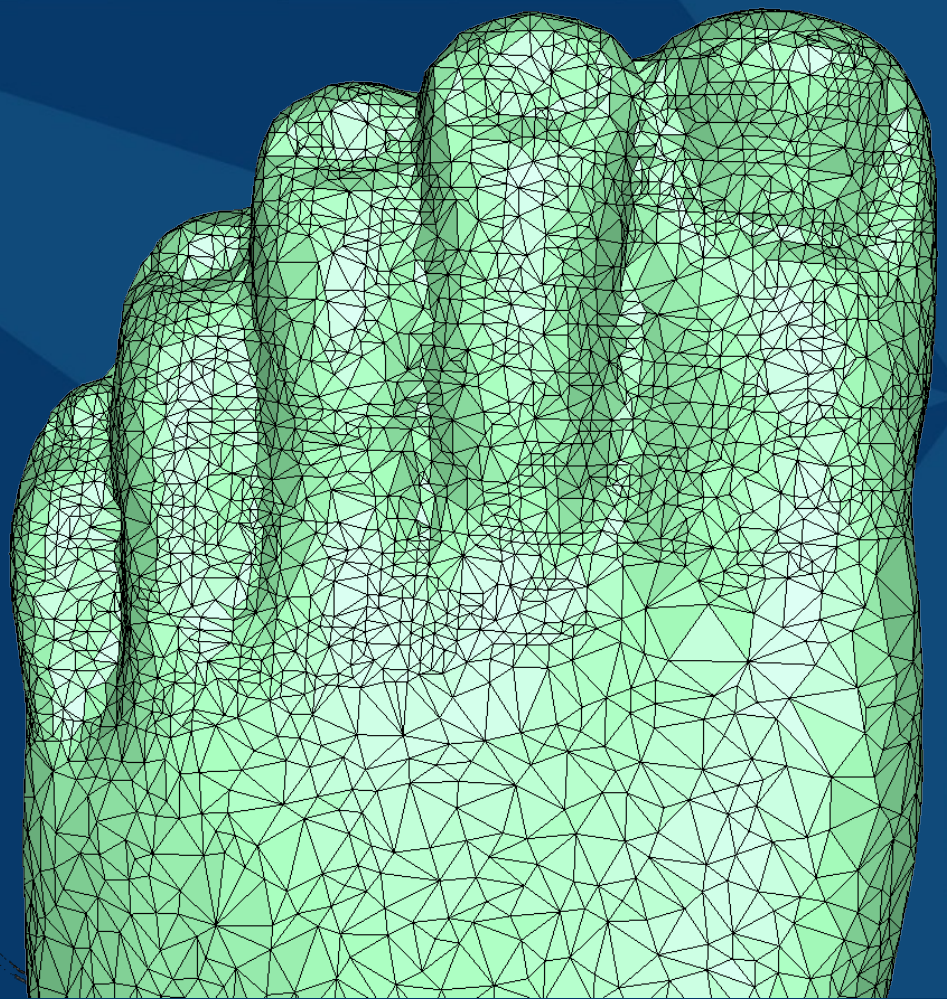


Remeshing from Head...

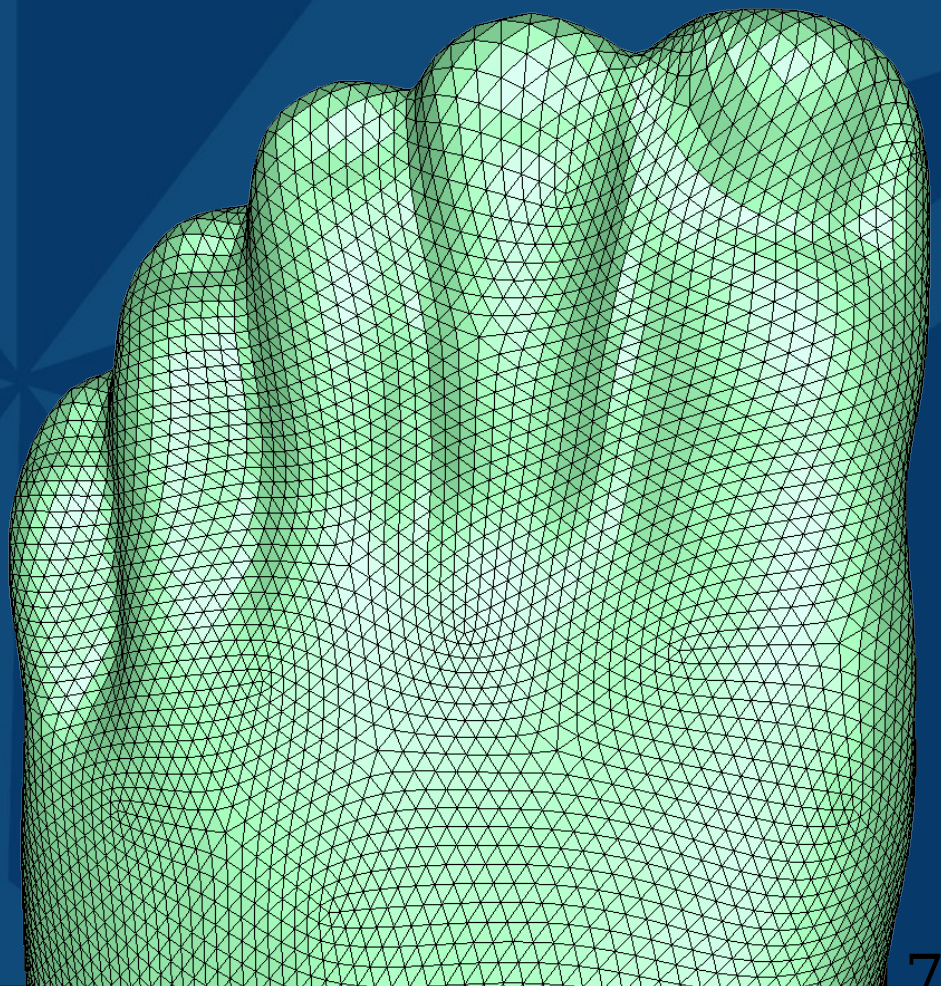


... to Toe

original mesh



uniform coarse remeshing,
then 2 levels of subdivision



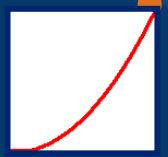
Other things you can do

Using normal map

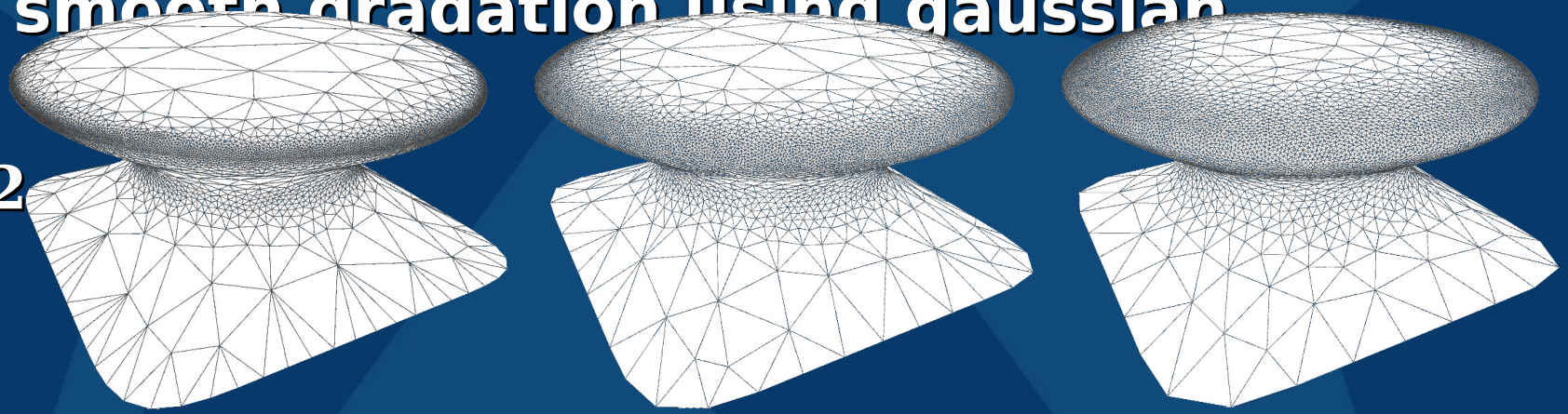
- back face culling
- silhouette enhancing (increase importance)
- extrusion, etc.

Using curvature map

- smooth gradation using gaussian

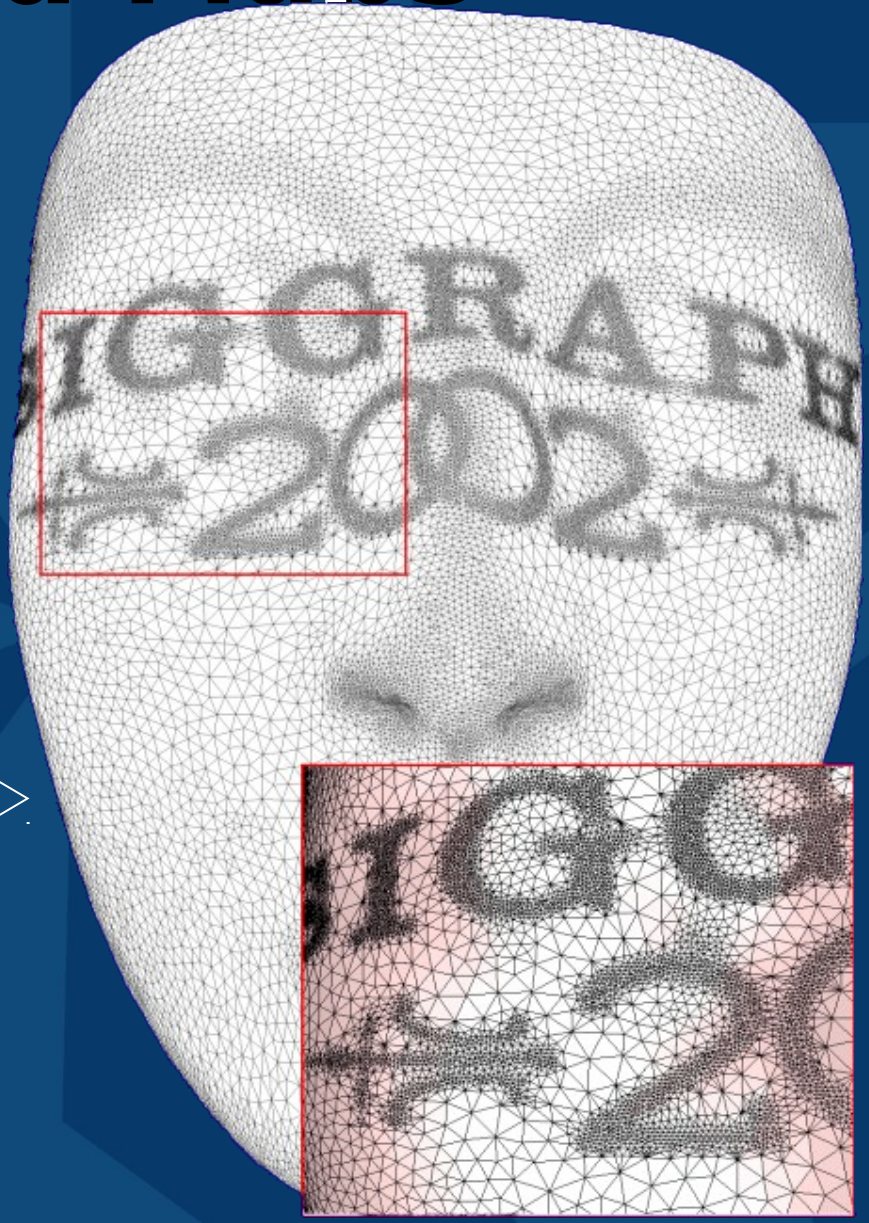


$\gamma = 2$



User-Defined Maps

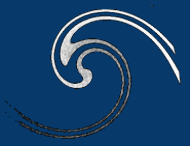
paint either in the importance map or directly on the mesh



Conclusion

Interactive Geometry Remeshing

- Area-balanced atlas
- Easy, rapid, and flexible design using 2D maps
- Real-time resampling
- Interactive, output-sensitive remeshing
- Correct handling of features and borders



Future Work

- **Improve precision**

  large images needed for high accuracy

- **Compression**

  rate/distortion approach

- **Approximation Theory**

  optimal sampling?

  with respect to what norm?